

	Type	L #	Hits	Search Text	DBs	Time Stamp
9	BRS	L25	2982	(707/2,3,4,8,104).ccls.	USPAT; Derwen t; IBM TDB	2000/11/02 08:34
10	BRS	L26	4	(707/2,3,4,8,104).ccls. and 119	USPAT; Derwen t; IBM TDB	2000/11/02 08:42
11	BRS	L27	36	(707/2,3,4,8,104).ccls. and (decision adj2 tree)	USPAT; Derwen t; IBM TDB	2000/11/02 08:42
12	BRS	L28	0	(707/2,3,4,8,104).ccls. and (decision adj2 tree) and dss	USPAT; Derwen t; IBM TDB	2000/11/02 08:42
13	BRS	L29	29	(707/2,3,4,8,104).ccls. and dss	USPAT; Derwen t; IBM TDB	2000/11/02 08:43
14	BRS	L30	16	(707/2,3,4,8,104).ccls. and dss and event	USPAT; Derwen t; IBM TDB	2000/11/02 08:43
15	BRS	L31	10	(707/2,3,4,8,104).ccls. and dss and event and constraints	USPAT; Derwen t; IBM TDB	2000/11/02 08:44
16	BRS	L32	5	(707/2,3,4,8,104).ccls. and dss and event and constraints and historical	USPAT; Derwen t; IBM TDB	2000/11/02 08:44

(FILE 'HOME' ENTERED AT 10:01:51 ON 02 NOV 2000)

FILE 'COMPENDEX, COMPUAB, COMPUSCIENCE, ELCOM, IFIPAT, INFODATA, INSPEC, INVESTEXT, JICST-EPLUS, MATH, MATHDI, NLDB, NTIS, PROMT, SCISEARCH, USPATFULL' ENTERED AT 10:01:59 ON 02 NOV 2000

L1 10 S SKEEN, MARION?/AU

L1 ANSWER 1 OF 10 IFIPAT COPYRIGHT 2000 IFI  
AN 3216399 IFIPAT;IFIUDB;IFICDB  
TI APPARATUS AND METHOD FOR PROVIDING DECOUPLED DATA COMMUNICATIONS BETWEEN  
SOFTWARE PROCESSES; COMPUTER PROGRAM  
IN Bowles Mark; **Skeen Marion Dale**  
PA Reuters Ltd GB  
PI US 5966531 19991012  
AI US 1995-478225 19950607  
RLI US 1990-632551 19901221 CONTINUATION 5557798  
US 1989-386584 19890727 CONTINUATION-IN-PART 5187787  
US 1990-601117 19901022 CONTINUATION-IN-PART 5257369  
FI US 5966531 19991012  
US 5557798  
US 5187787  
US 5257369  
DT UTILITY  
FS ELECTRICAL  
MRN 008728 MFN: 0434  
008729 0393  
CLMN 23  
GI 20 Drawing Sheet(s), 28 Figure(s).

L1 ANSWER 3 OF 10 IFIPAT COPYRIGHT 2000 IFI  
AN 2519398 IFIPAT;IFIUDB;IFICDB  
TI APPARATUS AND METHOD FOR CREATION OF A USER DEFINABLE VIDEO DISPLAYED  
DOCUMENT SHOWING CHANGES IN REAL TIME DATA  
IN Risberg Jeffrey S; **Skeen Marion D**  
PA Unassigned Or Assigned To Individual (68000)  
PI US 5339392 19940816 (CITED IN 112 LATER PATENTS)  
AI US 1990-636044 19901228  
RLI US 1989-386584 19890727 CONTINUATION-IN-PART 5187787  
US 1990-601117 19901022 CONTINUATION-IN-PART 5257369  
US 1990-632551 19901221 CONTINUATION-IN-PART  
FI US 5339392 19940816  
US 5187787  
US 5257369  
DT UTILITY; REASSIGNED; EXPIRED  
FS ELECTRICAL  
MRN 005649 MFN: 0007  
CLMN 46

**WEST****End of Result Set**

Generate Collection

L9: Entry 1 of 1

File: USPT

Aug 16, 1994

DOCUMENT-IDENTIFIER: US 5339392 A

TITLE: Apparatus and method for creation of a user definable video displayed document showing changes in real time data

## DEPR:

Referring to FIG. 4, there is shown a typical example of an "object" as that term is used in the object oriented programming world. An object is an entity which has both properties and associated operations which can be invoked by a user to change the values of various properties of the object. In the simple example chosen, the object is an entity representing a class of objects in the form of bank accounts. The particular instance in this class is a bank account for Jessica Doe. The properties of each object in this class or the account number, the balance and the owner name. The operations that can be invoked are withdrawal, deposit and query. Each operation can be invoked by entering a command to identify the object and start the operation (typically the name of the operation) followed by the argument. The operation then uses the argument to change the properties of the object. For example, the command Doe. Deposit(100) would invoke the deposit operation and add 100 units to the balance property of the Doe bank account.

## DEPR:

Sends the string `msg` to the server used by `rstream`. The messages are passed directly to the server, and are not in any way affected by the state of the stream. The messages are understood by the standard market data servers include "rr" to re-request a page, and "q a" to request the server's network address. Some messages induce a response from the server (such as queries). In this case, the response will be delivered to all streams that are connected to the server.

## DETL:

MDP.sub.-- MSG.sub.-- BAD = -1 MDP.sub.--  
MSG.sub.-- DATA = 0 Page data message. MDP.sub.-- MSG.sub.-- STATUS = 1  
Status/error message. MDP.sub.-- MSG.sub.-- OOB = 2 "Out of Band" message, e.g.,  
time stamp. MDP.sub.-- MSG.sub.-- QUERY = 3 Query result.

**WEST****End of Result Set**

Generate Collection

L14: Entry 1 of 1

File: USPT

Aug 16, 1994

DOCUMENT-IDENTIFIER: US 5339392 A

TITLE: Apparatus and method for creation of a user definable video displayed document showing changes in real time data

**BSPR:**

In the preferred embodiment, the tools available for defining an active document are as follows. A label tool allows the user to enter static text to label or annotate the active document or to create his or her own personalized help screens. A quote tools displays the value of an issue, including a user defined set of other fields pertaining to that particular company in a display style specified by the user. For example, a brief style displays only the price where a comprehensive style displays all the available fields. A ticker tool can be used as a selective or block ticker, and can show data in any display style. Upticks and Downticks can be shown in color and volume information can be included. A page fragment tool displays a region of a page-based feed such as Telerate or Reuters. Any region of the page designated by the user can be displayed from a single character to a full page. Highlighting modes are provided to highlight that has changed. A time based graph tool can be used to create graph display objects to graphically display the changes in value of a variable such as price per time. The time and price axes may be scaled to minutes or seconds, and the price value may be set to any unit such as 1/8 or 1/32 of a dollar. Above and below channel segments can be drawn on the graph. Graphs may be merged to show two issues against the same time axis. A data set graph tool can be used to create graph display objects which display the values of multiple instruments such as stocks or bonds or other subscribed values in real me such as a yield curve in a semiconductor processing application environment. Graphs may be merged to show two different sets of issues against each other to indicate market opportunities. A table tool can be used to create display objects which show position blotters, currency lookup tables, and names of commonly used pages securities. A publisher tool publishes information constructed using the invention or entered the user onto the network using the network communication process running in the environment in the invention is running. The published information can be used by other processes linked to the network or as a bulletin board for use by her traders. A button tool can be used to create splay objects that execute scripted actions when the button is "pushed", i.e., selected in any way on the splay such as by clicking on the button by a mouse. The scripted actions are entered by the user in whatever sequence is desired in a language such as the MarketScript.TM. command language comprised of all commands at the invention can execute. In the preferred embodiment, the scripted command sequence can also include commands to the operating system, the network communication software and other processes running on the same host or elsewhere on network. Buttons can be programmed to carry out commonly performed operations such as moving quickly to an important page or performing an operation to be carried out when an alert condition occurs. The buttons allow creation of hypertext links between different sheets, and the alert scripts can perform operations such as changing a color, flashing an object, sounding an audible alarm or executing an external program. The latter capability provides great flexibility by enabling clients to program features such as the ability to telephone a beeper service when an alert occurs. A "glossary" facility allows users to add new operations, i.e., commands, to the scripting language and customize the menus as needed.

**DEPR:**

Pops up a dialog box containing a list of numeric fields in records for this market type. Used to select which field to use in alert calculations. This dialog box also contains areas to enter four different scripts, which are executed under

different conditions, as explained below.

DEPR:

The Active Objects such as those shown at 66 consist of labels, quotes, tickers, buttons, graphs etc. Label objects are "active" even though they are text because they can, for example, be changed as part of a script of commands which are executed as a result of an alarm condition.

DEPR:

Referring to FIG. 7, there is shown a diagram of the parts of an Active Object. An instance of an Active Object 100 contains a data object 102, a Style Map 104, a Display Object 106, one or more Event Triggers shown generally at 108 and one or more Event Scripts of commands to be performed sequentially if an Event Trigger condition occurs. These Event Scripts are shown generally at 110. The Active Object 100 includes a local event dispatcher which dispatches events, e.g., user commands, to the various Active Object components for processing. For example, data updates from a network or database are dispatched to the Data Object 102, and "display frame" or display related events are dispatched to the Display Object 106. Events handled by the local dispatcher can be internally generated, such as a "change notification" event from the Data Object, which is dispatched to the Display Object so that the displayed representation can be updated.

DEPR:

The Event Trigger is a specification of conditions under which the user wishes to do extra processing on the Active Object. For example, the user can set alarm limits such as a certain price or trading volume for a particular quote Active Object, and when a real time data update indicates that the limit has been exceeded, an alarm condition exists to transfer the Active Object from the normal update state to the alert state. The Event Script of commands to execute upon occurrence of the specified alarm condition is specified in the Event Script specifications shown generally at 110. The things that can be scripted to happen upon occurrence of an alarm condition are limited only by the imagination of the user. Minimally, the script may specify an audible beep and/or a change in color of an Active Object. More exotic scripts may issue commands on the network to start another process running to dial a beeper, issue a sell order, issue a buy order, etc. Other scripts may publish some or all the data on one or more sheets of an active document on the network, etc. The commands in the scripting language generally include all the commands understood by the script processor as well as commands defined by the user and can, in some embodiments, include commands to the operating system, the high level network interface or other processes running on the network. Generally the commands understood by the script processor will include the name of the object, the desired operation and an argument, i.e., what value to set etc.

DEPR:

Further, the DCC library may change the communication protocol being used based upon changing conditions such as number of subscribers. For example, an intelligent Multicast protocol may be chosen (described in more detail below). In this protocol, a point-to-point protocol is used when the number of subscribers is below a certain cutoff number (programmable by the system administrator) but switch-over to a broadcast protocol automatically occurs when the number of subscribers rises above the cutoff number. In the preferred embodiment "high-water" and "low-water" marks are used as will be described below. In other embodiments, any cost function may be used to set the switch-over point based upon cost and efficiency of sending multiple point-to-point messages as opposed to a single broadcast message.

DEPR:

Step 928 is a test which checks if the quote object has received and stored a data form pointer, and whether the stylemap pointer to the new style map is valid (i.e., the stylemap name was one for which a stylemap exists). If these conditions are not both met, then step 930 repaints the background display of the quote object, to clear any prior invalid real-time content from the screen. Thereafter, processing returns to the calling routine. Typically, the MarketSheet software according to the TIB.RTM. software has a main loop and the processes symbolized by FIGS. 44 through 47A and 47B represent subroutines. As such the blocks labelled "return" in these figures represent returns to the place in the main loop where processing left off when the particular subroutine in question

was called.

DEPR:

If the conditions in step 928 are both met, then in step 932 the filtration specified by the style map identified in the change in stylemap name received in step 926 must be performed on the stored received data form. Thereafter, the selected data remaining after the filtration process must be displayed in the manner specified by the new style map and any data type conversions between the data type of the incoming data and the data type desired for display must be performed. These conversions are performed with the aid of the metadata taken from the newly arrived data aggregate, i.e., form. Block 932 represents the process of calling the subroutine that does this filtration and conversion processing. Block 932 actually represents a subroutine call to the subroutine "Style Map Apply" which is detailed in the flow chart of FIGS. 47A and 47B. In calling the "Style Map Apply" routine, a pointer to the newly arrived data aggregate and the style map ID of the style map selected by the user is passed to the "Style Map Apply" routine as arguments in the subroutine call.

CLPR:

28. The apparatus of claim 27 further comprising means for receiving data updates for a corresponding item of real time data, and for updating the corresponding real time data with its new the update value and storing the updated value in said display object means in accordance with said style data stored in said style map record, and for comparing any updated values for said real time data to any event trigger condition data stored in said means for storing event trigger condition data, and for displaying any special effects on said display according to said style data if any event trigger condition has been satisfied by an update to a corresponding item of real time data.

CLPR:

35. The apparatus of claim 34 wherein said program in execution includes a script processor program which causes said computer to execute a script comprised of a series of commands selected by said user when a predetermined trigger condition programmed by said user occurs.

CLPR:

36. The apparatus of claim 35 wherein said program in execution includes means for displaying icons which, when selected by said user using an input device, set a trigger condition and for linking the trigger condition associated with each said icon to a corresponding script such that when said icon is selected, said script of commands is executed by said computer.

CLPR:

37. The apparatus of claim 35 wherein said program in execution includes means for setting alarm limits for the value of any user selected item of real time data, the value of said alarm limit being selectable by said user, and for setting a predetermined trigger condition associated with each said alarm limit when said alarm limit is equalled or exceeded by the value of the corresponding item of real time data, and for selectively linking each said trigger condition to a corresponding script.

CLPV:

event trigger stored in said computer specifying the conditions under which the user desires additional processing by said computer to occur on this particular active object, typical event triggers being alarm limits, and

CLPV:

means for storing user defined event trigger condition data specifying the conditions for data stored in said data object record under which said user desires further action to occur; and

CLPV:

means coupled to said means for storing event trigger condition data, for storing one or more scripts of actions programmed by said user and associated with one or more events said actions of each script to be carried out by said computer in a sequence programmed by said user if a condition specified in the corresponding event trigger occurs.

**WEST**[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Posting Counts](#)[Show S Numbers](#)[Edit S Numbers](#)[Preferences](#)**Search Results -**

Terms	Documents
(5339392).pn. and conditions	1

**Database:**

US Patents Full-Text Database	▲
JPO Abstracts Database	
EPO Abstracts Database	
Derwent World Patents Index	
IBM Technical Disclosure Bulletins	▼

**Refine Search:**

(5339392).pn. and conditions

[Clear](#)**Search History****Today's Date: 11/2/2000**

<u>DB Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
USPT	(5339392).pn. and conditions	1	<a href="#">L14</a>
USPT	(5339392).pn. and restrict	0	<a href="#">L13</a>
USPT	(5339392).pn. and rules	0	<a href="#">L12</a>
USPT	(5339392).pn. and constraint	0	<a href="#">L11</a>
USPT	(5339392).pn. and contraint	0	<a href="#">L10</a>
USPT	(5339392).pn. and query	1	<a href="#">L9</a>
USPT	(5339392).pn. and (database or data adj base)	1	<a href="#">L8</a>
USPT	(5339392).pn. and sass	1	<a href="#">L7</a>
USPT	(5339392).pn. and service adj1 discipline	1	<a href="#">L6</a>
USPT	(5339392).pn. and service adj1 disciple	0	<a href="#">L5</a>
USPT	(5339392).pn. and support	1	<a href="#">L4</a>
USPT	(5339392).pn. and decision	1	<a href="#">L3</a>
USPT	(5339392).pn.	1	<a href="#">L2</a>
USPT	(5966531 or 5339392).pn.	2	<a href="#">L1</a>

	Type	L #	Hits	Search Text	DBs	Time Stamp
1	BRS	L17	186	decision adj2 support adj2 system	USPAT; Derwen t; IBM TDB	2000/11/02 08:27
2	BRS	L18	73890	117 and query and event constraint	USPAT; Derwen t; IBM TDB	2000/11/02 08:28
3	BRS	L19	27	117 and query and event and constraint	USPAT; Derwen t; IBM TDB	2000/11/02 08:28
4	BRS	L20	0	117 and complex adj2 query and event and constraint	USPAT; Derwen t; IBM TDB	2000/11/02 08:28
5	BRS	L21	2	117 adj2 query and event and constraint	USPAT; Derwen t; IBM TDB	2000/11/02 08:31
6	BRS	L22	5	skeen-marion\$.in.	USPAT; Derwen t; IBM TDB	2000/11/02 08:32
7	BRS	L23	0	skeen-marion\$.in. and decision adj2 support	USPAT; Derwen t; IBM TDB	2000/11/02 08:33
8	BRS	L24	0	(707/2,3,4,8, "104").ccls.	USPAT; Derwen t; IBM TDB	2000/11/02 08:34



**WEST**[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Posting Counts](#)[Show S Numbers](#)[Edit S Numbers](#)[Preferences](#)**Search Results -**

Terms	Documents
(5339392).pn. and optimize	1

**Database:**

US Patents Full-Text Database ▲

JPO Abstracts Database

EPO Abstracts Database

Derwent World Patents Index

IBM Technical Disclosure Bulletins ▼

**Refine Search:**

(5339392).pn. and optimize ▲▼

[Clear](#)**Search History****Today's Date: 11/2/2000**